FIG. 1A

BUFFER

F_1: STORE A POINTER TO NEXT BUFFER
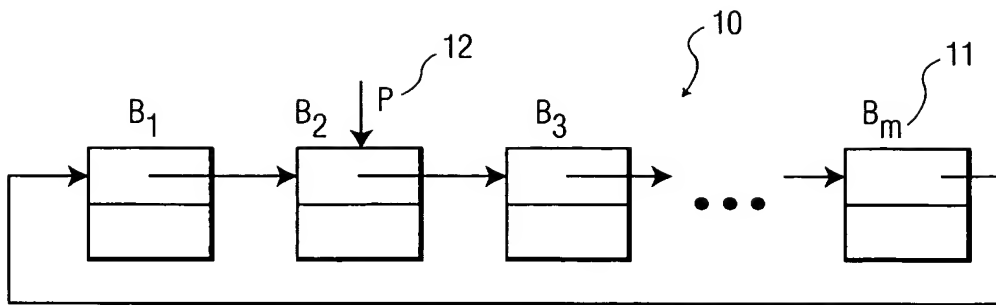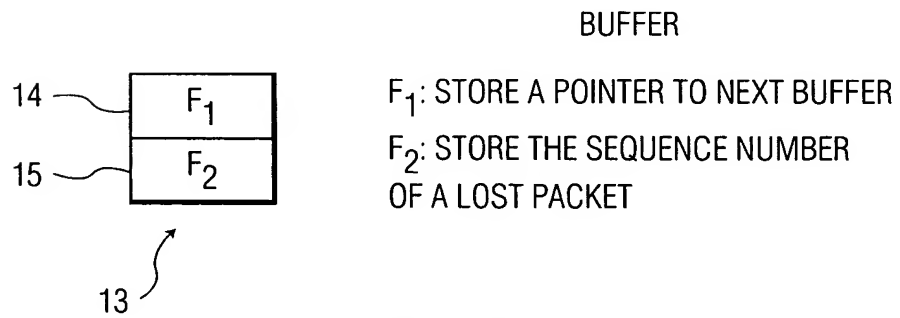
F_2: STORE THE SEQUENCE NUMBER
OF A LOST PACKET

FIG. 1B

CIRCULAR BUFFER INITIALIZATION:

$B_i \rightarrow F_2 = 0$, FOR i = 1,2,...m

$P = B_1$

NORMAL OPERATION:

VARIABLE:

s: CURRENT PACKET SEQUENCE NUMBER

h: LENGTH OF OBSERVED HOLE IN SEQUENCE NUMBER

P: THE POINTER THAT CIRCULATES THE CIRCULAR BUFFER

UPON **RECEIVING A PACKET** WITH SEQ. NO. x

```
{
        h=x-s;
        if (h>0)
        {
                if (h=1)  //NO HOLE
                {
                        if (P→F₂ ≠ 0) {
                                declare the packet with the seq. No of P→F₂ lost;
                                P→F₂ = 0;
                        }
                        P= P→F₁;  //MOVE THE POINTER TO NEXT BUFFER
                }

                while (h>1)
                {
                        if (h>1) //HAS HOLE
                        {
                                if (P→F₂ ≠ 0) {
                                        declare the packet with the seq. No of P→F₂ lost;
                                }
                                P→F₂ = ++s;  //A POSSIBLE LOSS
                                P = P→F₁;  //MOVE THE POINTER TO NEXT BUFFER
                        }
                        h--;
                }
                s=x;  //UPDATE THE CURRENT SEQUENCE NUMBER
        } else //RECEIVE AN OUT OF ORDER PACKET
        {
                find out Bᵢ that Bᵢ→F₂ =x, do
                        Bᵢ→F₂ =0; //CLEAN THE RECORD.
        }
}
```

FIG. 2A

FIG. 2B

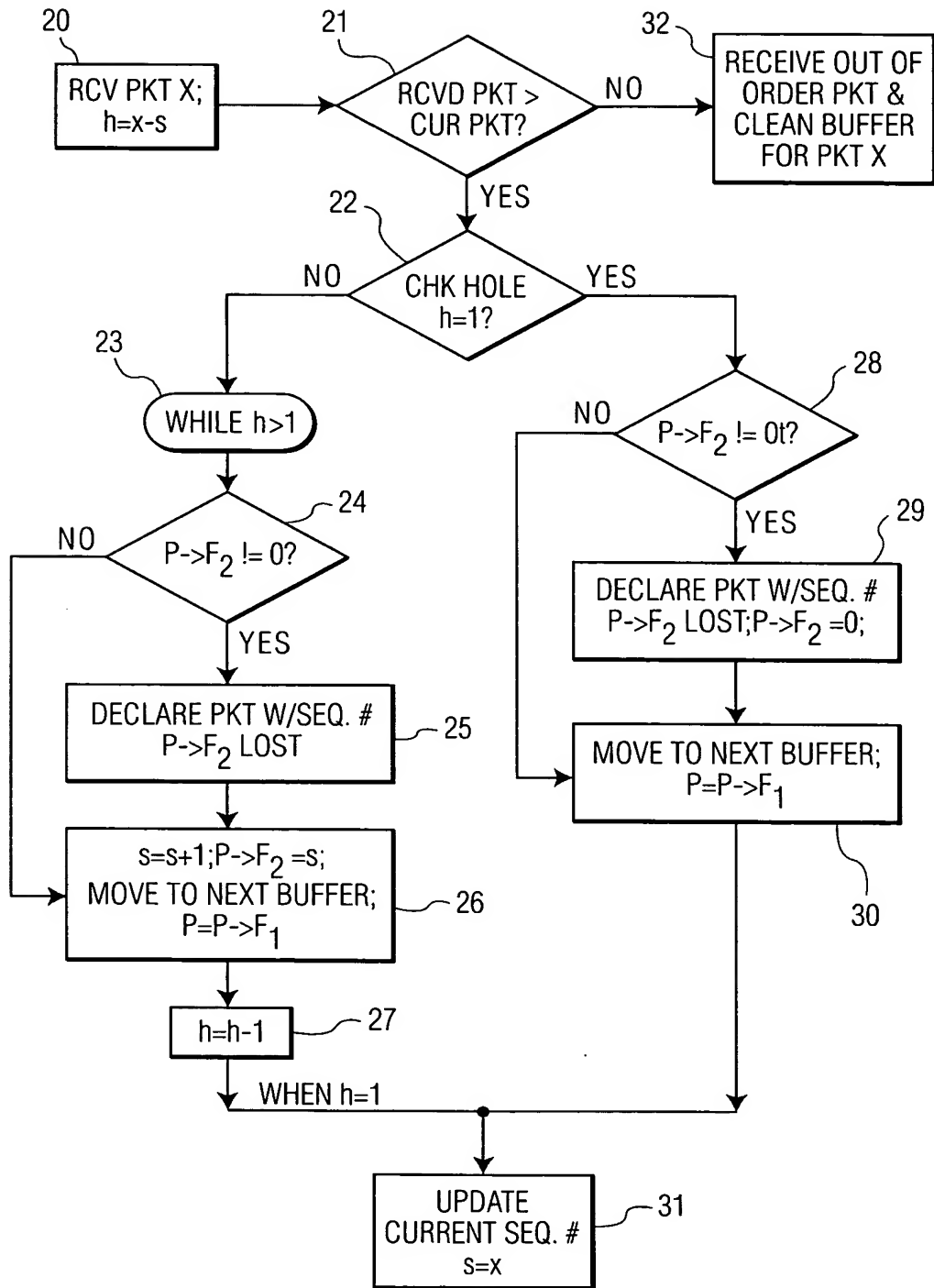**chain_size adaptation:**

INITIALIZE:
m=4; //INITIAL CHAIN SIZE
success_rate=EXPECT_RATE;   //DECLARED SUCCESS RATE
      //DEFINED AS THE RATIO OF TRUE LOSSES TO TOTAL DECLARED LOSSES
false_rate=0;   //FALSE DECLARATION RATE
      //DEFINED AS THE RATIO OF FALSELY DECLARED LOSSES TO TOTAL DECLARED LOSSES

UPON **RECEIVING A PACKET OR MAKING A LOSS DECLARATION**
{
    update success_rate and false_rate;
    if MIN_CHAIN_SIZE < m < MAX_CHAIN_SIZE   35
    if success_rate > EXPECT_RATE
        consider **deleting a buffer** from the chain and m--;
      if false_rate > TOLERABLE_RATE
                            36

FIG. 3